Method of Pre-processing a Deep Neural Network for Addressing Overfitting and Designing a Very Deep Neural Network

Dr. Nuriel S. Mor
nuriel.mor@gmail.com
+972523572145

Clinical psychology, Ph.D.
Applied psychology, M.Sc.
Medical science (pharmacology & physiology), M.Sc.
Computer science, M.Sc.
Computer science, B.Sc.
Neuroscience (psychology & biology), B.Sc.
Mathematics, B.Ed.

Abstract

Sparseness of hidden unit activation is the common effect of the three primary methods (unsupervised pre-training, rectifier neural networks, and dropout) that significantly reduce overfitting in deep neural networks (DNN) and improve their performance in discriminative tasks. Sparsity allows designing a complex DNN, enjoying the benefits of such an expressive model while at the same time mitigating the undesired effect of complexity. Only a small subset of units is active for each different input. Sparse coding in the brain, the neurophysiological equivalent of sparseness of hidden unit activation, allows precise discrimination between similar stimuli. In the brain, synaptic pruning is a mechanism which encourages sparse coding and essential for learning. In addition, Hebbian unsupervised learning is followed by synaptic pruning: synapses that are frequently active together have strong connections between them and are maintained, while the rarely used synapses with weak connections are eliminated (pruned). Hebbian learning rule is the underlying principle of unsupervised learning in DNN. The purpose of this article is to suggest a novel method which is consistent with these neuroscience observations and the order observed in the brain. The method includes the following steps: 1) start with a very big fully connected DNN with hundreds of nonlinear hidden layers. 2) perform fast unsupervised pre-training. 3) prune weights with the smallest absolute value. 4) perform the backpropagation algorithm on the trimmed DNN. Besides of the direct effects of this method on sparsity and on undesired complexity that causes overfitting. This technique is a strategy to design very deep neural networks with hundreds of nonlinear hidden layers while maintaining a reduced number of parameters overall in the backpropagation algorithm. Network depth is of crucial importance, and this method allows designing a very deep network by keeping a reduced number of parameters. Inspired by the human brain, very deep models with hundreds of hidden layers can move forward the field of artificial intelligence to a new level of performance. This suggested method is consistent not only with the order observed in the brain of unsupervised learning followed by synaptic pruning but also with the dynamic structural feature of the brain: starting with a very high connectivity and deliberately removing weak synapses, to further improve a particular functionality of the brain.

Introduction

Deep neural networks (DNNs) are composed of many non-linear hidden layers, and this makes DNN expressive models that can potentially learn very complicated relationships between their inputs and outputs [13, 23]. For decades, however, DNNs could not be trained to produce practical results because of overfitting [1]. The overfitting problem is increasingly likely to occur as the complexity of the DNN increases [8, 23]. If the DNN has enough hidden units to model complicated relationships between its inputs and outputs, there will be many different settings of the weights that can model perfectly the relationship in the training data [13]. Each of these weight vectors, however, will produce different predictions on test data and will do worse on the test data compared to the training data. The weights were adjusted to work well together on the training data but not on test data [13]. In other words, the DNN memorizes the training data and cannot generalize to new examples and this called overfitting.

Since the development of the backpropagation algorithm (learning algorithm for DNN) in 1986, many methods have been developed for reducing overfitting, but they failed in solving this problem in a satisfactory way that would have made DNN practical [1,23]. These include early stopping, weight decay, weight sharing, and model averaging. Only in 2006, researchers discovered a method that improves the performance of DNN [1,14,15] significantly. The technique is unsupervised pre-training using contrastive divergence and greedy wise layer learning [9,14,15]. The weights of the DNN are initialized in the unsupervised pre-training, and further fine tuning using the backpropagation algorithm should be used to improve the model for classification [14,15]. Until 2017, two additional main methods were shown to significantly reduce overfitting: deep sparse rectifier neural networks and dropout [1,6,10,11,17,21,30]. Different theoretical explanations were suggested to explain the efficiency of each of these three

methods. The efficacy of unsupervised pre-training was explained by suggesting that the input vectors contain more information than the labels, and the precious information in the labels only used for the fine tuning [14,15]. The efficacy of deep sparse rectifier neural networks was explained by advantages of sparsity [10]. The effectiveness of dropout was explained by preventing co-adaptation of weights [13]. However, there is one common effect to all these three methods: the activations of the hidden units become sparse [10,19,23]. For deep sparse rectifier neural networks, this effect stems directly from the properties of the rectifier. Unsupervised pre-training and dropout, however, are not direct sparsity-inducing regularizes, and still, they encourage sparseness of hidden unit activation [19,23]. Sparsity allows designing a very deep and complex DNN and enjoying the benefits of such an expressive model while at the same time mitigating the undesired effect of complexity. A very small and different subset of units is active for each different input.

Sparse Coding

Sparse coding is a fundamental neuroscience observation and has become a concept of interest in computational neuroscience [7,10]. Sparse coding refers to the neurobiochemical phenomenon that each stimulation is encoded by activation of a small set of neurons. For each stimulation to be encoded, this is a different subset of all available neurons [7,10]. It was introduced in computational neuroscience in the context of sparse coding in the visual, auditory, touch, and olfactory systems [5,16,20,29]. Researchers [5,16,20,29] claimed that sparse coding is the computational mechanism allowing precise discrimination between similar stimuli. Researchers [10] suggested that sparse coding enhances the capacity to of the biological brain to perform discriminative tasks by reducing overlap between representations. This neuroscience observation are relevant to the performance of DNNs in discriminative tasks [10]. The fact that

the three primary methods that successfully reduce overfitting induce sparseness of hidden unit activation is in accordance with this computational neuroscience principle of sparse coding.

The purpose of this paper is to suggest a pre-processing procedure for DNN that is based on a neurobiochemical process that induces sparsity in the biological brain. A process of synaptic pruning.

Synaptic Pruning

Synaptic pruning refers to the neurobiochemical process in the brain which includes deliberate synapse elimination [2,3]. This process takes place in mammals' brain mainly between early childhood and the onset of puberty [2,4]. Researchers claimed that synaptic pruning is influenced by learning and is claimed to represent and support learning [2,3,4,22]. At birth, the brain starts with a very high connectivity, and during childhood and adolescence, in the process of synaptic pruning, weak synapses are deliberately removed in order to further improve a particular network capacity and a specific functionality in the brain.

Synaptic pruning is determined by synaptic plasticity principles, and Hebbian learning rule: synapses that are frequently active together have strong connections between them and are maintained while the rarely used synapses with weak connections are eliminated. Researchers argued that synaptic pruning removes unnecessary neuronal structures from the brain, reduces undesired redundant complexity from the brain, to support further learning [2,3,22]. Important to notice the order of synaptic plasticity and Hebbian learning rule followed by synaptic pruning. First, Hebbian learning rule determines the synaptic strength and then synaptic pruning according to the strength of synapses determined by Hebbian learning rule. As mentioned, in DNN, the overfitting problem is the result of increased complexity of DNN. Therefore, synaptic

pruning enhances learning through two mechanisms: inducing sparsity and removing unnecessary neuronal structures.

Weight Pruning

Weight pruning is an attempt to implement synaptic pruning in DNN [18,27,28]. Researchers [18,27,28] found that weight pruning improves performance of DNNs but not significantly enough to solve the overfitting problem. Weight pruning has been implemented during the backpropagation algorithm. This article suggests a novel way to implement weight pruning, **before** the backpropagation algorithm, that is consistent with neuroscience observations. The suggestion is weight pruning following unsupervised training and before the backpropagation algorithm, in order to support learning in subsequent discriminative learning in the backpropagation algorithm.

Method Description

Method of Pre-processing a Deep Neural Network Which Includes Fast Unsupervised Pre-Training Combined with Weight Pruning Before Backpropagation For Addressing Overfitting and Designing a Very Deep Neural Network

As mentioned, in the brain, synaptic plasticity principle of Hebbian learning rule followed by synaptic pruning. Hebbian learning rule is the underlying principle of unsupervised learning in deep belief networks and deep sigmoid belief networks [9,14,15]. Therefore, the suggestion of this article is to follow these synaptic plasticity principles and apply it in DNN. The suggestion is to start with a very big fully connected network with multiple nonlinear hidden layers and parameters. Perform fast unsupervised pre-training. Prune weights with the smallest absolute value. Perform backpropagation algorithm on the trimmed DNN. This suggested

procedure is consistent with neuroscience observations of an unsupervised learning process followed by synaptic pruning to support further learning. The effects of this procedure are sparseness of hidden unit activation because of the pruning, and removal of undesired complexity of the DNN before the backpropagation algorithm. As mentioned, sparseness of hidden unit activation is the common effect of the three primary methods that successfully address overfitting, and complexity of DNN causes overfitting in the backpropagation algorithm.

This suggested pre-processing procedure is consistent with neuroscience observations of synaptic pruning following unsupervised learning in the mammals' brain. This pre-processing procedure induces sparsity and reduces undesired complexity of DNN which are essential in solving overfitting.

In addition, this suggested pre-processing procedure allows to design DNN with hundreds of hidden layers and maintain a reduced number of paraments present in the backpropagation algorithm, and by that this method helps to avoid overfitting while enjoying the benefits of a very deep network. Such a DNN can be a very expressive model because of multiple non-linear hidden layers but with a reduced number of parameters overall, because of the pruning. Network depth is of crucial importance, and very deep models can be very beneficial [11,12]. Inspired by the human brain, very deep models with hundreds of hidden layers can move forward the field of artificial intelligence to a new level of performance.

This method suggested in the article is consistent not only with the order observed in the brain of unsupervised learning followed by synaptic pruning but also with the structural dynamic feature of the brain. Starting at birth, with a very high connectivity and deliberately removing weak synapses determined earlier by an unsupervised Hebbian learning rule, to further improve a particular network capacity and a specific functionality of the brain

References

[1] Ba, J., & Frey, B. (2013). Adaptive dropout for training deep neural networks. In *Advances in Neural Information Processing Systems* (pp. 3084-3092).

[2] Craik, F. I., & Bialystok, E. (2006). Cognition through the lifespan: mechanisms of change. *Trends in cognitive sciences*, *10*(3), 131-138.

[3] Chechik, G., Meilijson, I., & Ruppin, E. (1999). Neuronal regulation: A mechanism for synaptic pruning during brain maturation. *Neural Computation*, *11*(8), 2061-2080.

[4] Chechik, G., Meilijson, I., & Ruppin, E. (1998). Synaptic pruning in development: a computational account. *Neural computation*, *10*(7), 1759-1777.

[5] Crochet, S; Poulet, JFA; Kremer, Y; Petersen, CCH (2011). "Synaptic mechanisms underlying sparse coding of active touch". *Neuron*. 69: 1160–1175.

[6] Dahl, G. E., Sainath, T. N., & Hinton, G. E. (2013). Improving deep neural networks for LVCSR using rectified linear units and dropout. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 8609-8613). IEEE.

[7] Dayan, P., & Abbott, L. F. (2001). Theoretical neuroscience (Vol. 806). Cambridge, MA: MIT Press.

[8] Erhan, D., Manzagol, P. A., Bengio, Y., Bengio, S., & Vincent, P. (2009). The difficulty of training deep architectures and the effect of unsupervised pre-training. In *International Conference on artificial intelligence and statistics* (pp. 153-160).

[9] Geoffrey E. Hinton (2007) Boltzmann machine. Scholarpedia, 2(5):1668.

[10] Glorot, X., Bordes, A., & Bengio, Y. (2011, June). Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (pp. 315-323).

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision* (pp. 1026-1034).

[13] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*

[14] Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, *18*(7), 1527-1554.

[15] Hinton, G. E. (2007). To recognize shapes, first learn to generate images. *Progress in brain research*, *165*, 535-547.

[16] Hromádka, T; Deweese, MR; Zador, AM (2008). "Sparse representation of sounds in the unanesthetized auditory cortex". PLoS Biol. 6: e16.

[17] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

[18] Lazarevic, A., & Obradovic, Z. (2001). Effective pruning of neural network classifier ensembles. In *Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on* (Vol. 2, pp. 796-801). IEEE.

[19] Li, J., Luo, W., Yang, J., & Yuan, X. (2013). Unsupervised Pretraining Encourages Moderate-Sparseness. *arXiv preprint arXiv:1312.5813*.

[20] Lin, Andrew C., et al. "Sparse, decorrelated odor coding in the mushroom body enhances learned odor discrimination." Nature neuroscience 17.4 (2014): 559-568.

[21] Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In *Proc. ICML* (Vol. 30, No. 1).

[22] Paolicelli, R. C., Bolasco, G., Pagani, F., Maggi, L., Scianni, M., Panzanelli, P., ... & Ragozzino, D. (2011). Synaptic pruning by microglia is necessary for normal brain development. *science*, *333*(6048), 1456-1458.

[23] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting.*The Journal of Machine Learning Research*, *15*(1), 19

[27]Thimm, G., & Fiesler, E. (1997). *Pruning of neural networks* (No. EPFL-REPORT-82417). IDIAP.29-1958.

[28] Tsodyks, M. V., & Feigel'Man, M. V. (1988). The enhanced storage capacity in neural networks with low activity level. *EPL (Europhysics Letters)*, *6*(2), 101.

[29] Vinje, WE; Gallant, JL (2000). "Sparse coding and decorrelation in primary visual cortex during natural vision". *Science*. 287: 1273–1276

[30] Wan, L., Zeiler, M., Zhang, S., Cun, Y. L., & Fergus, R. (2013). Regularization of neural networks using dropconnect. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)* (pp. 1058-1066).